

University of Groningen

Computing normalizing transformations

Broer, H.; Hoveijn, I.; Lunter, G.; Vegter, G.

Published in:
EPRINTS-BOOK-TITLE

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2003

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Broer, H., Hoveijn, I., Lunter, G., & Vegter, G. (2003). Computing normalizing transformations. In *EPRINTS-BOOK-TITLE* University of Groningen, Johann Bernoulli Institute for Mathematics and Computer Science.

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

7 Computing normalizing transformations

Given a singularity and a versal deformation of it, an arbitrary deformation can be induced from the versal one by a smooth transformation of coordinates and parameters. Kas and Schlessinger [KS72] developed an algorithm to compute this transformation, for deformations of functions. It requires a procedure to solve the infinitesimal stability equation, which is provided by the normal form map NF^Ψ related to a standard ideal basis (see Chap. 6). A similar approach works for the case of deformations of maps, involving left-right tangent spaces instead of ideals.

7.1 Introduction

In Chap. 5 we quoted a result from Mather that may be summarized as follows:

A deformation is versal if it is transversal.

The condition of transversality is an *infinitesimal* condition. It is expressed in terms of the tangent space to the orbit of the undeformed singularity, under the group of transformations considered. If this tangent space has finite codimension (in the appropriate function space), and moreover the deformation directions span the quotient space, the deformation is called transversal. In contrast, versality is a *local* condition. A deformation is versal if it can be connected to an arbitrary deformation of the same singularity through smooth reparametrizations and coordinate transformations, on a full neighborhood in parameter- and phase-space. This local property immediately implies the infinitesimal condition. Mather's reverse implication however is a deep result.

The algorithm of Kas and Schlessinger may be viewed as a first step in the direction of Mather's result. From the infinitesimal or linearized condition, it gives a method of computing the formal power series solution of the required reparametrization and coordinate transformation. Mather's result implies existence of such a formal solution, but gives no algorithm to compute it. On the other hand, the existence of a formal solution does not imply existence of a smooth solution in any neighborhood, the existence proof of which involves the Mather-Malgrange preparation theorem.

The main ingredient in Kas and Schlessinger's algorithm is a procedure to express any (truncated) formal power series into a 'tangent' part, and a 'transversal' part in some fixed finite-dimensional vector space with a dimension equal to the codimension of the singularity's tangent space. For deformations of functions under right-equivalences, this tangent space is an ideal. Given a standard basis of the ideal, the division algorithm is precisely such a procedure. It splits a truncated formal power series in an element of the ideal, the tangent part, plus a unique rest-term playing the role of the 'transversal' part.

In the case of maps, and left-right transformations, the tangent space has a different structure, but essentially the same ideas apply. The structure of the tangent space (for both cases) is described in Chap. 5. The machinery for computing standard bases for ideals and left-right tangent spaces is described in Chap. 6. The current chapter puts these results together and describes in detail how these are used to compute formal reparametrizations and coordinate transformations, up to any desired order.

Section 7.2 deals with the relatively easy case of functions orbiting under the group of right-transformations. Section 7.3 deals with maps under left-right transformations.

Base ring and formal power series Some of the results in this chapter are formulated in terms of the ring \mathcal{E}_n of germs of functions $\mathbb{R}^n \rightarrow \mathbb{R}$. Later we also use rings of formal power series $\mathbb{R}[[x]]$. Both are abstractions of the ring used for actual computations, namely truncated formal power series (over a computable field; see also the remarks in Chap. 6). To avoid excessive notation, we shall in this chapter not explicitly use this truncated ring. As long as a graded term ordering is used, all statement for the full power series ring continue to hold in the truncated setting. See also Sect. 6.5.3.

7.2 Deformations of functions

The problem dealt with in Sect. 2.2.6 can be stated as follows. Suppose a deformation $G(x, v)$ of a function $f(x) = G(x, 0)$ is given. Here $x \in \mathbb{R}^n$ can be thought of as phase space variables, and $v \in \mathbb{R}^c$ as (small) parameters. Now consider the following two problems:

1. Produce a universal deformation $F(x, u) : \mathbb{R}^{n+d} \rightarrow \mathbb{R}$ of $f(x)$, and
2. Compute a reparametrization $h : \mathbb{R}^q \rightarrow \mathbb{R}^d$ and coordinate transformation $\phi : \mathbb{R}^{n+q} \rightarrow \mathbb{R}^n$ inducing G from F .

(See Sect. 5.3 for definitions.) One could say that (1) gives a catalog of possible bifurcations, and (2) gives an index into this catalog for the given family G . The first problem amounts to computing a basis for the vector space $\mathcal{E}_n/J(f)$, where \mathcal{E}_n is the ring of germs of functions on \mathbb{R}^n , and $J(f)$ is the Jacobian ideal,

$$J(f) = \left\langle \frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right\rangle_{\mathcal{E}_n}.$$

The hypothesis that such a (finite) basis for $\mathcal{E}_n/J(f)$ exists is equivalent to requiring that $J(f)$ has finite codimension. Under this hypothesis, the basis elements can be computed by reducing to truncated formal power series. Chapter 6 describes algorithms to compute an analogue of Gröbner bases for ideals in truncated formal power series rings. Using these bases, called *standard bases*, generators of $\mathcal{E}_n/J(f)$ are easily found.

Suppose $t_1(x), \dots, t_d(x) \in \mathcal{E}_n$ are such generators. Another way of putting this is to say that the equation

$$(7.1) \quad g(x) = \alpha_1(x) \frac{\partial f}{\partial x_1} + \dots + \alpha_n(x) \frac{\partial f}{\partial x_n} + \mu_1 t_1(x) + \dots + \mu_d t_d(x)$$

has a solution in $\alpha_i \in \mathcal{E}_n$ and $\mu_i \in \mathbb{R}$, for arbitrary $g \in \mathcal{E}_n$. Equation (7.1) is called the *infinitesimal stability equation*. Problem 2 is reduced, by an algorithm of Kas and Schlessinger [KS72], to solving (7.1) a number of times, in the ring of truncated formal power series. The standard basis of $J(f)$ is again used to compute the α_i and μ_i efficiently.

We now go into a bit more detail.

7.2.1 Finding a universal deformation

We start by recalling a few results and definitions from Chap. 5, in particular Sect. 5.3.1. A deformation F (of f) is called *versal* if any other deformation G (of f) can be induced from it, and *universal* if it has the minimum possible number of deformation parameters. It is called *transversal* if the ‘initial speeds’ (see [Mar82]) or ‘deformation directions’ $\{\frac{\partial F}{\partial u_1}|_{u=0}, \dots, \frac{\partial F}{\partial u_d}|_{u=0}\}$ complement the tangent space $T_f = J(f)$. By theorem 5.5, a deformation F is versal if and only if it is transversal, reducing the problem of finding a universal deformation of f to finding complements of the ideal $J(f)$ as a vector subspace of \mathcal{E}_n .

Proposition 7.1. *Let $f \in \mathcal{E}_n$ be a germ of some function, and suppose that $J(f) \supseteq \mathfrak{m}^p$, where \mathfrak{m} is the maximal ideal of \mathcal{E}_n . Let $<$ be a term order with respect to which \mathfrak{m}^p is a truncation ideal. Let $\{h_1, \dots, h_k\}$ be a standard basis w.r.t. $<$ of the projection of $J(f)$ in the ring of truncated formal power series. Then the following monomials forms a basis for $\mathcal{E}_n/J(f)$:*

$$\{m : m \notin \langle \text{LM } h_1, \dots, \text{LM } h_k \rangle\}.$$

Suppose the monomials are labeled t_1, \dots, t_d , then a universal deformation of f is given by

$$(7.2) \quad F(x, u) = f(x) + u_1 t_1(x) + \dots + u_d t_d(x).$$

Proof: Because $\{h_1, \dots, h_k\}$ is a standard basis, the t_i complement $\text{LM}(J(f) + \mathfrak{m}^p)$. Since $\mathcal{E}_n/\mathfrak{m}^p$ is finite dimensional, this implies that $\mathbb{R}\{t_1, \dots, t_d\} + J(f) + \mathfrak{m}^p = \mathcal{E}_n$, and using $J(f) \supseteq \mathfrak{m}^p$ also $\mathbb{R}\{t_1, \dots, t_d\} + J(f) = \mathcal{E}_n$, proving the first

statement. Versality of F follows from theorem 5.5. If one from $\{t_1, \dots, t_d\}$ is removed, the remaining ones do not complement $J(f)$, so F cannot be versal. This proves universality. ■

In practice the truncation-order p may not be known beforehand. One chooses a large p and checks afterwards whether $\mathfrak{m}^{p-1} \subseteq J(f)$.

7.2.2 The algorithm of Kas and Schlessinger

We now address the question of computing the transformations that induce arbitrary deformations from a universal one. Suppose $F(x, u) : \mathbb{R}^{n+d} \rightarrow \mathbb{R}$ and $G(x, v) : \mathbb{R}^{n+c} \rightarrow \mathbb{R}$ are deformations of f , and suppose F is (uni)versal. The goal is to find a parameter-dependent coordinate transformation $\phi : \mathbb{R}^{n+q} \rightarrow \mathbb{R}^n$, and a reparametrization $h : \mathbb{R}^q \rightarrow \mathbb{R}^d$, satisfying

$$(7.3) \quad \begin{aligned} \phi(x, 0) &= x, \\ h(0) &= 0, \\ G(x, v) &= F(\phi(x, v), h(v)). \end{aligned}$$

Kas and Schlessinger's algorithm [KS72] accomplishes this, in the ring of truncated formal power series. The idea is to expand ϕ and h with respect to the total degree of the *parameters* v , and solve iteratively for increasing order in v . The details are as follows. Define

$$\phi(x, v) := \sum_{i \geq 0} \phi_i(x, v), \quad h(v) := \sum_{i \geq 0} h_i(v),$$

where ϕ_i and $h_i(v)$ are homogeneous of degree i in v , and denote the partial sums up to total order p in v by superscripting with the degree:

$$\phi^p(x, v) := \sum_{i=0}^p \phi_i(x, v), \quad h^p(v) := \sum_{i=0}^p h_i(v).$$

Now assume that (7.3) has been solved up to order p , that is,

$$(7.4) \quad G(x, v) = F(\phi^p(x, v), h^p(v)) + O(v^{p+1}) + O(x^m)$$

(where it is supposed that we truncate at order m in x). This equation holds for $p = 0$ if we define $\phi^0(x, v) = x$ and $h^0(v) = 0$, since by assumption $F(x, 0) = G(x, 0) = f(x)$. To solve (7.3) up to order $p + 1$ we add $(p + 1)$ st order terms in v :

$$\begin{aligned} F(\phi^{p+1}, h^{p+1}) &= \\ F(\phi^p + \phi_{p+1}, h^p + h_{p+1}) &= \\ F(\phi^p, h^p) + D_x F(\phi^p, h^p) \cdot \phi_{p+1} + D_u F(\phi^p, h^p) \cdot h_{p+1} + O((\phi_{p+1} + h_{p+1})^2) &= \\ F(\phi^p, h^p) + D_x f(\phi^p, h^p) \cdot \phi_{p+1} + D_v F(\phi^p, h^p)|_{v=0} \cdot h_{p+1} + O(v^{p+2}). \end{aligned}$$

To obtain the last equality we used the estimates $\phi^p(x, v) = x + O(v)$, $\phi_{p+1}(x, v) = O(v^{p+1})$, $h^p(v) = O(v)$ and $G(x, v) = f(x) + O(v)$. Now we plug in the explicit form (7.2) for F . Writing $h_{p+1,k}$ and $\phi_{p+1,k}$ for the k -th components of the $p+1$ -st order terms of h and ϕ , the equation to be solved for (7.3) to hold up to order $p+1$ is

$$(7.5) \quad G(x, v) - F(\phi^p(x, v), h^p(v)) = \sum_{k=1}^n \phi_{p+1,k}(x, v) \frac{\partial f}{\partial x_k} + \sum_{k=1}^d h_{p+1,k}(v) t_k(x) + O(v^{p+2}).$$

By (7.4), the left-hand-side does not contain terms of order less than $p+1$. Equation (7.5) can be solved by equating coefficients of terms v^σ left and right, where $|\sigma| = \sigma_1 + \dots + \sigma_d = p+1$. For each term v^σ we obtain an equation of the form (7.1), the infinitesimal stability equation. Since $F(x, v)$ is supposed to be a universal deformation, these equations can be solved. This proves existence of a formal power series for ϕ and h (or a solution up to any desired order in x and v) solving (7.3).

The algorithm of Kas and Schlessinger is a recipe for computing ϕ and h , given a procedure for solving (7.1). In practice it is important to be able to solve (7.1) efficiently, as for every term v^σ one such equation is encountered. A procedure that meets this criterion is described in the next section.

Remark 7.2. (*Parameters vs. phase variables*) It is important to note that the transformations obtained converge as power series in the *parameters* v , and not necessarily in the *phase variables* x . Even if a power series solution in x exists (i.e., no zeroth-order terms are needed), then Kas and Schlessinger's algorithm may not find it, because of the non-uniqueness of the solutions $\alpha_i(x)$ in (7.1).

7.2.3 Solving the infinitesimal stability equation

The problem of computing the transformations inducing a given deformation from a versal one is now reduced to solving (7.5), i.e., (7.1), a number of times. We re-state the problem slightly using the notation of Chap. 6. Write $h_i := \frac{\partial f}{\partial x_i}$, suppose $\langle h_1, \dots, h_n \rangle$ has finite codimension, and let t_1, \dots, t_d denote the monomials not in $\text{LM} \langle h_1, \dots, h_n \rangle$. Then $\langle h_i \rangle + \mathbb{R}\{t_1, \dots, t_d\} = \mathcal{E}_n$, in other words, for any $g \in \mathcal{E}_n$ it is possible to solve

$$(7.6) \quad g = \sum_{i=1}^n \alpha_i h_i + \sum_{i=1}^d \mu_i t_i$$

for $\alpha_i \in \mathcal{E}_n$ and $\mu_i \in \mathbb{R}$.

The procedure is as follows. First add new elements $h_i \in \langle h_1, \dots, h_n \rangle$ ($i > n$) to the ideal basis, using Buchberger's algorithm (see Sect. 6.2.4), until the basis $\{h_1, \dots, h_n, h_{n+1}, \dots, h_k\}$ is a standard basis. The output r of the *reduced*

normal form algorithm 6.14 of Sect. 6.3.7 is an element of $\mathbb{R}\{t_1, \dots, t_d\}$, that is, can be written as $r = \sum_{i=1}^d \mu_i t_i$. Then, the following equation holds:

$$g = \sum_{i=1}^k \alpha_i h_i + \sum_{i=1}^d \mu_i t_i,$$

where α_i are also output of the reduced normal form algorithm. This equation is, apart from the upper limit in the first sum, of the form (7.6). So if only the numbers μ_i are required, this solves our problem.

If it is also required to express the ideal member $g - r$ in terms of the h_1, \dots, h_n , we can use the extended Buchberger algorithm. To write down this algorithm, we need some notation:

Definition 7.3.

- a) $M_k := \bigoplus_{i=1}^k \mathcal{E}_n e_i$, the free k -dimensional \mathcal{E}_n -module,
- b) $\Psi_k : M_k \rightarrow \mathcal{E}_n : f e_i \mapsto f h_i$ ($i = 1, \dots, k$), an \mathcal{E}_n -module homomorphism,
- c) $\Theta_k : M_k \rightarrow M_n$, an \mathcal{E}_n -module homomorphism.

Here the e_i are just symbols, and the h_i are elements of \mathcal{E}_n . The algorithm below also uses the notation s_{ij} for the element of M_k (with $k \geq i, j$) whose Ψ -image is the S-polynomial of h_i and h_j . Note that $\text{NF}_\alpha^{\Psi_k}$ maps \mathcal{E}_n into M_k . With these notations, the algorithm is the following:

Algorithm 7.4. (Extended Buchberger algorithm)

Input: $h_1, \dots, h_n \in \mathcal{E}_n$

Output: $h_1, \dots, h_k \in \mathcal{E}_n$ and an \mathcal{E}_n -module homomorphism $\Theta : M_k \rightarrow M_n$, with the properties:

1. $\{h_1, \dots, h_k\}$ is a standard basis for $\langle h_1, \dots, h_n \rangle$.
2. $\Psi_k \alpha = \Psi_n \Theta \alpha$, for any $\alpha \in M_k$.

Algorithm:

```

 $k \leftarrow n$ 
 $\Theta_k \leftarrow$  identity map
While  $\text{NF}_r^{\Psi_k}(\Psi_k s_{ij}) \neq 0$  for any  $1 \leq i < j \leq k$ , do:
     $h_{k+1} \leftarrow \text{NF}_r^{\Psi_k}(\Psi_k s_{ij})$ 
     $\Theta_{k+1}|_{M_k} \leftarrow \Theta_k$ , and
     $\Theta_{k+1} e_{k+1} := \Theta_k(s_{ij} - \text{NF}_\alpha^{\Psi_k}(\Psi_k s_{ij}))$ 
     $k \leftarrow k + 1$ 
EndWhile
 $\Theta \leftarrow \Theta_k$ 

```

In an implementation, the map Θ is easily represented by an $n \times k$ matrix of elements of \mathcal{E}_n .

Proof: For the proof of termination, and of the fact that $\{h_1, \dots, h_k\}$ forms a standard basis, see the proof of algorithm 6.17 in Sect. 6.4.1. To prove that

$\Psi_k \alpha = \Psi_n \Theta_k \alpha$ for all k , we proceed by induction. The assertion is trivial for $k = n$, so assume it holds for some k . The definition of Θ_{k+1} assures that it maps into M_n . By hypothesis, $\Psi_{k+1}|_{M_k} = \Psi_n \Theta_{k+1}|_{M_k}$, so it remains to show that $\Psi_n \Theta_{k+1} e_{k+1} = \Psi_{k+1} e_{k+1}$:

$$\begin{aligned}
 (7.7) \quad \Psi_n \Theta_{k+1} e_{k+1} &= \Psi_n \Theta_k (s_{ij} - \text{NF}_\alpha^{\Psi_k}(\Psi_k s_{ij})) = \\
 &\Psi_k (s_{ij} - \text{NF}_\alpha^{\Psi_k}(\Psi_k s_{ij})) = \\
 &\Psi_k s_{ij} - \Psi_k \text{NF}_\alpha^{\Psi_k}(\Psi_k s_{ij}) = \\
 &\text{NF}_r^{\Psi_k}(\Psi_k s_{ij}) = h_{k+1} = \Psi_{k+1} e_{k+1},
 \end{aligned}$$

where in the first step the definition of Θ_{k+1} , and in the second step the induction hypothesis was used. \blacksquare

With the map Θ , solving the infinitesimal stability equation (7.6) becomes trivial: Given $g \in \mathcal{E}_n$, compute $\alpha := \Theta \text{NF}_\alpha^\Psi(g)$ and $\mu := \text{NF}_r^\Psi(g)$, then write them in the form

$$\alpha = \alpha_1 e_1 + \cdots + \alpha_n e_n, \quad \mu = \mu_1 t_1 + \cdots + \mu_d t_d,$$

with $\alpha_i \in \mathcal{E}_n$ and $\mu_i \in \mathbb{R}$. Now $\alpha_1, \dots, \alpha_n, \mu_1, \dots, \mu_d$ solve (7.6).

7.2.4 Application: The hyperbolic umbilic

Here we apply the method of the previous section to the singular germ $x(x^2 + y^2)$. This singularity is commonly known as the hyperbolic umbilic, or D_4^+ in Arnol'd's classification [Arn81]. For this singularity, the general method above is like using a mirror to look at your bangles.¹ However, the underlying idea is the same as in the case of left-right tangent spaces, dealt with in Sect. 7.3, but without the technical complications, so it also serves as an introduction to the next section.

Let $f = x(x^2 + y^2)$, then the generators of $J(f)$ are $h_1 := \frac{\partial f}{\partial x} = 3x^2 + y^2$ and $h_2 := \frac{\partial f}{\partial y} = 2xy$. Take a graded term order with $y < x$, then the only syzygy is $s_{12} = 2ye_1 - 3xe_2$, with $\Psi s_{12} = 2y^3$, which is a monomial not in $\langle \text{LM } h_1, \text{LM } h_2 \rangle$. Therefore $\text{NF}_\alpha^{\Psi_2}(\Psi s_{12}) = 0$ and Θ_3 becomes

$$\Theta_3 e_i = e_i \quad (i = 1, 2) \quad \text{and} \quad \Theta_3 e_3 = 2ye_1 - 3xe_2.$$

New syzygies are $s_{13} = 2y^3 e_1 - 3x^2 e_2$ and $s_{23} = y^2 e_2 - x e_3$, and these are reduced to zero, proving that

$$\{3x^2 + y^2, 2xy, 2y^3\}$$

is a standard ideal basis for $J(f)$. Using algorithm 6.3.7 and the map Θ the infinitesimal stability equation can be solved efficiently.

¹ Hindi proverb

7.3 Deformations of maps

This section is the analogue of Sect. 7.2 for the case of deformations of the energy-momentum map, instead of the planar Hamiltonian function. Let $F(x, u) : \mathbb{R}^{n+d} \rightarrow \mathbb{R}^2$ be a deformation of $\mathbf{E}(x) := F(x, 0)$. Universality of these deformations is defined with respect to the class of left-right transformations

$$(A, B) : \quad \mathbf{E} \mapsto B \circ \mathbf{E} \circ A,$$

where $A : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $B : \mathbb{R}^2 \rightarrow \mathbb{R}^2$. Suppose that $G(x, v) : \mathbb{R}^{n+c} \rightarrow \mathbb{R}^2$ is any deformation of \mathbf{E} . Again there are the two basic problems: Finding a universal deformation F of \mathbf{E} , and computing a coordinate transformation connecting it to G . Their solution is also similar in spirit to the function-case. For functions, the tangent space was the Jacobian ideal $J(f)$. For maps, the algebraic structure of the tangent space is much more involved, and this results in technical complications. The machinery of Chap. 6 helps us out, however.

The energy-momentum map $\mathbf{E} : x \mapsto (H, H_2)$ which we are interested in, is of a special form. The first component is the Hamiltonian function, the second component is its quadratic part. Exploiting this, we arrived in Sect. 3.2.3 at the reduced tangent space:

$$T_{\mathbf{E}}^r = J + \{1, f_1, f_2\}\mathbb{R}[[H, H_2]].$$

Here the f_i are functions related to H and H_2 , and J is an ideal whose generators are given by equation (3.9). The ability to write any (truncated) formal power series as an element of $T_{\mathbf{E}}^r$ plus a rest-term in some finite dimensional vector space, is crucial to computing the coordinate transformation.

7.3.1 Adaptation of Kas and Schlessinger's algorithm

In this section we show how to compute the transformations inducing an arbitrary deformation from a universal one. So suppose $F(x, u) : \mathbb{R}^{n+d} \rightarrow \mathbb{R}^2$ is a universal deformation of \mathbf{E} , and let $G : \mathbb{R}^{n+q} \rightarrow \mathbb{R}^2$ be an arbitrary deformation of the same map. We are looking for a left-right transformation (A, B) , depending on parameters $v \in \mathbb{R}^q$, and a reparametrization $h : \mathbb{R}^q \rightarrow \mathbb{R}^d$, such that

$$\begin{aligned} A(x, 0) &= x, \\ B(y, 0) &= y, \\ h(0) &= 0, \\ (7.8) \quad G(x, v) &= B \left(F(A(x, v), h(v)), v \right) = B \circ F(\cdot, h) \circ A. \end{aligned}$$

We proceed in the same way as in Kas and Schlessinger's algorithm. Expand A , B and h with respect to the total order in the parameters v , and write

$$A(x, v) := \sum_{i \geq 0} A_i(x, v), \quad B(y, v) := \sum_{i \geq 0} B_i(y, v), \quad h(v) := \sum_{i \geq 0} h_i(v),$$

where A_i , B_i and h_i are homogeneous of degree i in v . Denote the partial sums up to and including order p by superscripting with p :

$$A^p(x, v) := \sum_{i=0}^p A_i(x, v), \quad B^p(y, v) := \sum_{i=0}^p B_i(y, v), \quad h^p(v) := \sum_{i=0}^p h_i(v).$$

Assume that A^p , B^p and h^p solve (7.8) up to order p . This is true for $p = 0$ if we set $A^0(x, v) = x$, $B^0(y, v) = y$ and $h^0(v) = 0$, since $F(x, 0) = G(x, 0) = \mathbf{E}(x)$. To solve (7.8) for the next order $p + 1$ we add $(p + 1)$ st order terms:

$$\begin{aligned} B^{p+1} \circ F(\cdot, h^{p+1}) \circ A^{p+1} &= (B^p + B_{p+1}) \circ F(\cdot, h^p + h_{p+1}) \circ (A^p + A_{p+1}) \\ &= B^p \circ F(\cdot, h^p) \circ A^p \\ &\quad + B_{p+1}(F(A^0(x, v), h^0(v)), v) \\ &\quad + B^0(D_u F(A^0(x, v), u)|_{u=0} \cdot h_{p+1}) \\ &\quad + B^0(D_x F(A^0(x, v), h^0(v))) \cdot A_{p+1} + O(|v|^{p+2}) \\ &= B^p \circ F(\cdot, h^p) \circ A^p \\ &\quad + B_{p+1}(\mathbf{E}(x), v) \\ &\quad + D_u F(x, u)|_{u=0} \cdot h_{p+1} \\ &\quad + D_x \mathbf{E}(x) \cdot A_{p+1} + O(|v|^{p+2}). \end{aligned}$$

This expression should be equal to $G(x, v)$ up to (but not including) order $p + 2$ in v . Since the term $B^p \circ F(\cdot, h^p) \circ A^p$ is already equal to $G(x, v)$ up to $O(|v|^{p+1})$ terms, the other three terms should account for the remaining terms in v of order $p + 1$. The resulting equation is

$$\begin{aligned} (7.9) \quad G(x, v) &- B^p \circ F(\cdot, h^p) \circ A^p \\ &= B_{p+1}(\mathbf{E}(x), v) + \sum_{i=1}^d (t_i(x), 0) \cdot h_{p+1, i}(v) + \alpha_{p+1}(x, v) \mathbf{E}(x) + O(|v|^{p+2}). \end{aligned}$$

Here we wrote α_{p+1} for a vector field on \mathbb{R}^n of order $p + 1$ in v , and we used that $F(x, u)$ is of the form

$$F(x, u) = \mathbf{E}(x) + u_1(t_1(x), 0) + \cdots + u_d(t_d(x), 0).$$

Note that (7.9) is an equation of *maps* to \mathbb{R}^2 .

From this point on, we use two facts that are particular to the application to the energy-momentum map. The first is that the projection of the tangent space $T_{\mathbf{E}}$ to its second component is surjective, so that deformation terms can be chosen of the form $(t_i, 0)$ (see Sect. 3.2.3). The second fact is more of a condition: We do not want to leave the circle-symmetric setting, in which we found ourselves after the Birkhoff procedure; therefore the vector field α is required to be circle-equivariant.

Collecting terms of the form v^σ of order $p + 1$, solving (7.9) boils down to solving, given $\mathbf{g}(x)$ and $\mathbf{E}(x)$ and $t_i(x)$, several instances of an equation of the form

$$(7.10) \quad \mathbf{g}(x) = \beta(\mathbf{E}(x)) + \alpha(x)\mathbf{E}(x) + \sum_{i=1}^d (t_i(x), 0)\mu_i$$

for maps $\beta : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, circle-equivariant vector fields α , and real numbers μ_i . From the assumption that $F(x, u) = \mathbf{E}(x) + \sum_i (t_i, 0)u_i$ is a universal deformation of $\mathbf{E}(x)$ it follows that this equation has a solution.

In Sect. 3.2.3, the tangent space $T_{\mathbf{E}}$ is reduced to

$$T_{\mathbf{E}}^r = J + \{1, f_1, f_2\}\mathbb{R}[[H, H_2]],$$

where $J = \langle h_0, h_1, h_2 \rangle$. With the results of Sect. 6.4.5 we can find functions t_1, \dots, t_d complementing $T_{\mathbf{E}}^r$. More precisely, given an element $g \in \mathbb{R}[[x]]$, there are $a_0, a_1, a_2 \in \mathbb{R}[[x]]$ and $\gamma_0, \gamma_1, \gamma_2 \in \mathbb{R}[[y_1, y_2]]$ and $\mu_1, \dots, \mu_d \in \mathbb{R}$ such that

$$(7.11) \quad g(x) = \sum_{i=0}^2 a_i h_i + \gamma_0(H, H_2) + f_1 \gamma_1(H, H_2) + f_2 \gamma_2(H, H_2) + \sum_{i=0}^d \mu_i t_i.$$

We shall refer to equation (7.11) as the *infinitesimal stability equation*, since it is the equivalent of (7.1) in the current context. Solving this equation is the subject of section 7.3.2 below. A solution to (7.11) translates, reading Sect. 3.2.3 backwards, into a solution for (7.10). This works as follows. For the notation, see Sect. 3.2.3.

Suppose the left-hand-side of (7.10) has the form

$$\mathbf{g}(x) = (g_1(x), g_2(x)).$$

Note that although at the first pass we have $g_2(x) \equiv 0$, since the second component of the unfolding $G(x, v)$ is constant, this may not be true at subsequent passes. Write

$$(7.12) \quad g_2(\rho_1, \rho_2, \psi) = a_0 + \rho_1 a_1(\rho_1, \rho_2, \psi) + \rho_2 a_2(\rho_1, \rho_2, \psi) + \psi a_3(\rho_1, \rho_2, \psi),$$

where $a_0 \in \mathbb{R}$. Using (3.4) or (3.5), we can find a vector field α' such that

$$(7.13) \quad \alpha' H_2 = g_2 - a_0.$$

In fact,

$$\begin{aligned} \alpha' := & \frac{a_1}{q} \left(\rho_1 \frac{\partial}{\partial \rho_1} + \frac{1}{2} p \psi \frac{\partial}{\partial \psi} \right) + \frac{a_2}{p} \left(\rho_2 \frac{\partial}{\partial \rho_2} + \frac{1}{2} q \psi \frac{\partial}{\partial \psi} \right) + \\ & a_3 \left(\frac{1}{2q} \psi \frac{\partial}{\partial \rho_1} + \frac{1}{2p} \psi \frac{\partial}{\partial \rho_2} + \rho_1^{p-1} \rho_2^{q-1} \left(\frac{p}{4q} \rho_2 + \frac{q}{4p} \rho_1 \right) \frac{\partial}{\partial \psi} \right) \end{aligned}$$

works. Now define

$$g := g_1 - \alpha'' H$$

and suppose a_i , γ_i and μ_i solve (7.11) for this g . Recall that the ideal generators h_i of J correspond to generators \mathbf{w}_i of the module of circle-equivariant vector fields that leave H_2 invariant:

$$\begin{aligned}\mathbf{w}_1 &= 2\rho_1\rho_2 \left(p \frac{\partial}{\partial \rho_1} - q \frac{\partial}{\partial \rho_2} \right) + (p^2\rho_2 - q^2\rho_1)\psi \frac{\partial}{\partial \psi}, \\ \mathbf{w}_2 &= 2\psi \left(p \frac{\partial}{\partial \rho_1} - q \frac{\partial}{\partial \rho_2} \right) + (p^2\rho_1^{p-1}\rho_2^q - q^2\rho_1^p\rho_2^{q-1}) \frac{\partial}{\partial \psi}.\end{aligned}$$

The generators of J are all defined in terms of the \mathbf{w}_i by

$$h_1 := \mathbf{w}_1 H, \quad h_2 := \mathbf{w}_2 H,$$

with the exception of h_0 , which is the relation. Also recall the definition of α_1 and α_2 (see Sect. 3.2.3). These vector fields obey $\alpha_1 H_2 = H$ and $\alpha_2 H_2 = H_2$, moreover $\alpha_1 H = -f_1$ and $\alpha_2 H = -f_2$. Using this, and the solution of (7.11) we define

$$\begin{aligned}\beta_1(y_1, y_2) &:= \gamma_0(y_1, y_2), \\ \beta_2(y_1, y_2) &:= y_1\gamma_1(y_1, y_2) + y_2\gamma_2(y_1, y_2) + a_0, \\ \alpha &:= a_1\mathbf{w}_1 + a_2\mathbf{w}_2 + \alpha' - \gamma_1(H, H_2)\alpha_1 - \gamma_2(H, H_2)\alpha_2.\end{aligned}$$

We then get

$$\alpha H_2 + \beta_2(H, H_2) = a_1\mathbf{w}_1 H_2 + a_2\mathbf{w}_2 H_2 + \alpha' H_2 + a_0 = g_2$$

(modulo h_0) since $\mathbf{w}_i H_2 = 0$ by definition of J , and

$$\begin{aligned}\alpha H + \beta_1(H, H_2) &= \\ a_1\mathbf{w}_1 H + a_2\mathbf{w}_2 H + \alpha' H - \gamma_1(H, H_2)\alpha_1 H - \gamma_2(H, H_2)\alpha_2 H + \gamma_0(H, H_2) &= \\ a_1 h_1 + a_2 h_2 + (g_1 - g) + \gamma_1(H, H_2)f_1 + \gamma_2(H, H_2)f_2 + \gamma_0(H, H_2) &= \\ g_1 - \sum \mu_i t_i(x) &\end{aligned}$$

(also modulo h_0), where we used (7.11). These calculations show that α and $\beta = (\beta_1, \beta_2)$ and μ_1, \dots, μ_d solve (7.10). Summarizing, we get the following:

Proposition 7.5. Suppose $\mathbf{E}(x) = (H(x), H_2(x))$ and $\mathbf{g}(x) = (g_1(x), g_2(x))$. Write g_2 as in (7.12) and let α' be such that (7.13) holds. Let $g := g_1 - \alpha' H$, and suppose that $a_i \in \mathbb{R}[[x]]$, $\gamma_i \in \mathbb{R}[[y]]$ and $\mu_i \in \mathbb{R}$ are such that (7.11) holds for this g . Then

$$\begin{aligned}\alpha &:= a_1\mathbf{w}_1 + a_2\mathbf{w}_2 + \alpha' - \gamma_1(H, H_2)\alpha_1 - \gamma_2(H, H_2)\alpha_2 \\ \beta(y_1, y_2) &:= (\gamma_0(y_1, y_2), y_1\gamma_1(y_1, y_2) + y_2\gamma_2(y_1, y_2) + a_0)\end{aligned}$$

together with the μ_i solve (7.10) modulo h_0 .

7.3.2 Solving the infinitesimal stability equation

The main building block of the algorithm for computing a left-right transformation inducing the given deformation $G(x, v)$ from the universal deformation $F(x, v)$, is the solution to the *infinitesimal stability equation* (7.11), as was shown in the previous section. If the generators $(\{h_i\}, \{f_i\}, \{g_i\})$ described in Proposition 3.3 form a standard basis of the reduced tangent space

$$T_{\mathbf{E}}^r = \langle h_0, h_1, h_2 \rangle + \{1, f_1, f_2\} \mathbb{R}[[g_1, g_2]],$$

the deformation directions t_i are easily found, and the normal form algorithm immediately gives a solution to (7.11).

However, these generators do *not* form a standard basis, and we should add generators to turn it into a one. This is done with a suitable generalization of Buchberger's algorithm, already outlined in Sect. 6.4.5. This creates a new problem: The normal form algorithm now returns a solution to the 'extended' infinitesimal stability equation, the equation with the new generators added to it. This solution cannot be used directly to build the left-right transformation. Just as in Sect. 7.2.3 this problem is solved by extending the Buchberger algorithm so that it computes a map Θ that maps expressions using standard basis generators, to those using only the original generators. In this way a solution to the extended infinitesimal stability equation is mapped to a solution to (7.11).

Now some remarks on Buchberger's algorithm for this case. Since $T_{\mathbf{E}}^r$ is an $\mathbb{R}[[g_1, g_2]]$ -module, to find a standard basis one can compute the normal forms of syzygies described by lemma 6.31, and add nonzero ones to the $\{f_i\}$ until each syzygy is reduced to 0. Though this algorithm is correct, efficient it is not. The reason is that it always adds nonzero syzygies at the 'coarsest' level of an $\mathbb{R}[[g_1, g_2]]$ -module, while for example ideal-syzygies, which are reduced *using ideal generators only* to nonzero normal form, may be added to the ideal generators $\{h_i\}$, thereby reducing the codimension of $\text{Im } \tilde{\Psi}$ at least as much, and possibly far more, than addition to the module generators $\{f_i\}$ would. Also, adding to the subalgebra-generators as much as possible leads to increased efficiency. An outline of the resulting algorithm is as follows:

- a) Expand $\{h_i\}$ to a standard ideal basis of $\langle h_i \rangle$, then
- b) Expand $\{g_i\}$ to a standard subalgebra basis of $\mathbb{R}[[g_i]]$, then
- c) Expand $\{f_i\}$ to make $(\{h_i\}, \{f_i\}, \{g_i\})$ a standard basis for $\langle h_i \rangle + \{f_i\} \mathbb{R}[[g_i]]$.

Remark 7.6. (*Efficiency*) One could add a stage before the last, namely extending $\{g_i\}$ to make $(\{h_i\}, \{g_i\})$ a standard basis for $\langle h_i \rangle + \mathbb{R}[[g_i]]$. This makes sense since $f_0 = 1$ and therefore syzygies for $\langle h_i \rangle + \mathbb{R}[[g_i]]$ are also syzygies for the left-right tangent space. However the practical improvement turns out to be slight in our case.

The map Θ , mapping back to original generators, is built up during the Buchberger algorithm. We now introduce some notation for the intermediate maps.

Note that we start counting from h_0 and f_0 , to be compatible with the notation used throughout.

Definition 7.7.

- a) $M_k := \bigoplus_{i=0}^k R e_{1i}$
- b) $N_{lm} := \bigoplus_{i=0}^l \mathbb{R}[[y_1, \dots, y_m]] e_{2i}$
- c) $\Theta_{klm} : M_k \oplus N_{lm} \rightarrow M_2 \oplus N_{22}$

Note that $T_{\mathbf{E}}^r$ has 3 ideal generators (including h_0), 3 module generators (including $f_0 = 1$) and 2 subalgebra generators. The maps Ψ_{klm} are defined, in Sect. 6.4.5, in terms of the basis $\langle h_0, \dots, h_k \rangle + \{f_0, \dots, f_l\} \mathbb{R}[[y_1, \dots, y_m]]$, and map from $M_k \oplus N_{lm}$ to R . On M_k these maps are R -module homomorphisms. On N_{lm} they are $\mathbb{R}[[y_1, \dots, y_m]]$ -module homomorphisms, with obvious multiplication in the domain, while on the range $y_i \cdot f := g_i f$. We also introduce the restricted maps $\Psi_k^h : M_k \rightarrow R$ and $\Psi_m^g : N_{0m} \rightarrow R$, with the superscripts denoting the generators in terms of which they are defined.

With this notation we can write down the extended Buchberger algorithm. Note that in the text below, whenever a new map $\Theta_{(k+1)lm}$ is defined, it is understood to coincide with the previous Θ_{klm} on $M_k \oplus N_{lm}$.

Algorithm 7.8. (*Extended Buchberger algorithm for LR-tangent spaces*)

Input: $h_0, h_1, h_2, f_0 = 1, f_1, f_2, g_1, g_2 \in R$

Output: Elements h_i, f_i, g_i and a map $\Theta_{klm} : M_k \oplus N_{lm} \rightarrow M_2 \oplus N_{22}$ such that

- a) $(\{h_0, \dots, h_k\}, \{f_0, \dots, f_l\}, \{g_1, \dots, g_m\})$ is a standard basis for the left-right tangent space $\langle h_i \rangle + \{f_i\} \mathbb{R}[[g_i]]$,
- b) $\Psi_{klm} \alpha = \Psi_{222} \Theta_{klm} \alpha$ for all $\alpha \in M_k \oplus N_{lm}$.

Algorithm:

$k \leftarrow 2, \quad l \leftarrow 2, \quad m \leftarrow 2$

$\Theta_{222} \leftarrow$ Identity map

While $\text{NF}_r^{\Psi_k^h}(\Psi_k^h(s_{ij})) \neq 0$ for any $1 \leq i < j \leq k$, do:

$h_{k+1} \leftarrow \text{NF}_r^{\Psi_k^h}(\Psi_k^h(s_{ij}))$

$\Theta_{(k+1)lm} e_{1(k+1)} := \Theta_{klm} \left(s_{ij} - \text{NF}_\alpha^{\Psi_k^h}(\Psi_k^h(s_{ij})) \right)$

$k \leftarrow k + 1$

EndWhile

Compute subalgebra-syzygies $\{b_i\}$ using Algorithm 6.24

While $\text{NF}_r^{\Psi_m^g}(\Psi_m^g(b_i)) \neq 0$ for any i , do:

$g_{m+1} \leftarrow \text{NF}_r^{\Psi_m^g}(\Psi_m^g(b_i))$

$\Theta_{kl(m+1)} y_{m+1} := \Theta_{klm} \left(b_i - \text{NF}_\alpha^{\Psi_m^g}(\Psi_m^g(b_i)) \right)$

$m \leftarrow m + 1$

Re-compute syzygies $\{b_i\}$ for basis $\{g_1, \dots, g_m\}$

EndWhile

Compute LR-tangent space syzygies (Lemma 6.31, 6.15; Alg. 6.26, 6.34, 6.36)

While $\text{NF}_r^{\Psi_{klm}}(\Psi_{klm}(S)) \neq 0$ for syzygy S of 2nd or 3rd kind, do:
 $f_{l+1} \leftarrow \text{NF}_r^{\Psi_{klm}}(\Psi_{klm}(S))$
 $\Theta_{k(l+1)m} e_{2(l+1)} := \Theta_{klm} (S - \text{NF}_\alpha^{\Psi_{klm}}(\Psi_{klm}(S)))$
 $l \leftarrow l + 1$
 Re-compute syzygies for basis $(\{h_1 \dots h_k\}, \{f_1 \dots f_l\}, \{g_1 \dots g_m\})$
 EndWhile

Proof: In general the output $\text{NF}_r^{\Psi}(f)$ is equal to f modulo $\text{Im } \Psi$. Since $f \in \text{Im } \Psi$ it follows that $h_{k+1} \in \langle h_1, \dots, h_k \rangle$, $g_{m+1} \in \mathbb{R}[[g_1, \dots, g_m]]$ and $f_{l+1} \in \{h_i\} + \{f_i\}\mathbb{R}[[g_i]]$. This with the fact that the condition in the final While-loop is false at exit proves part (a), since syzygies of the first kind reduce to zero as $\{h_1, \dots, h_k\}$ has been extended to a standard ideal basis in the first part of the algorithm.

For part (b) we show that $\Psi_{klm}\alpha = \Psi_{222}\Theta_{klm}\alpha$ is an invariant of the algorithm. Indeed, for $k = l = m = 2$ equality is trivial, and for other values invariance follows, by induction, from the equality $\text{NF}_r^{\Psi}(\Psi S) = \Psi(S - \text{NF}_\alpha(\Psi S))$, which holds for any S and any map Ψ for which a normal form is defined. This proves part (b).

Termination is guaranteed if we work inside the finite dimensional vector space of truncated formal power series, since each new generator increases the set $\langle \text{LM } h_i \rangle + \{\text{LM } f_i\}\mathbb{R}[[\text{LM } g_i]]$ by at least one monomial, by definition of the normal form. ■

7.3.3 Example of a LR-tangent space calculation

This section is intended to show how algorithm 7.8 works on an easy example. The example was chosen so as to exhibit the essential features of a generic problem instance, but has no extra ‘meaning’ by itself. We consider the following LR-tangent space:

$$(7.14) \quad T = \langle xy^2, x^2y + a_1y^4 + a_2x^4 \rangle + \{1, xy\}\mathbb{R}[[a_3(x^2 + y^2), a_4x^4 - a^5x^5]].$$

This is the image of $\Psi : M_2 \oplus N_{22} \rightarrow R$ as follows:

$$\begin{aligned} M_2 &= \{e_{11}, e_{12}\}R \\ N_{22} &= \{e_{21}, e_{22}\}\mathbb{R}[[y_1, y_2]] \\ \Psi : e_{11}r &\mapsto h_1r \quad (r \in \mathbb{R}) \\ e_{12}r &\mapsto h_2r \\ e_{21}Y &\mapsto f_1Y(g_1, g_2) \quad (Y \in \mathbb{R}[[z_1, z_2]]) \\ e_{22}Y &\mapsto f_2Y(g_1, g_2). \end{aligned}$$

Below we shall also use the restricted maps $\Psi^h : M_2 \rightarrow R$ and $\Psi^g : \mathbb{R}[[z_i]] \rightarrow R$.

The variables a_1, \dots, a_5 are coefficients, and are treated as constants. During the calculation certain nondegeneracy-conditions in terms of these coefficients are encountered. These can be read off from the final results.

Standard ideal basis Following algorithm 7.8, first we find a standard ideal basis for $\langle xy^2, x^2y + a_1y^4 + a_2x^4 \rangle$. As a term order we choose the graded order with $x < y$. The first (and only) syzygy between the generator's leading monomials is

$$s_{12} = xe_{11} - ye_{12}$$

whose image under Ψ^h is $-a_2yx^4 - a_1y^5$. This element reduces to $-a_1y^5 + a_2a_1x^2y^4 + a_2^2x^6$ by addition of a_2x^2 times the second generator. The result has leading monomial y^5 , which is not in $\text{Im } \tilde{\Psi}^h = \langle xy^2, x^2y \rangle$. Adding it to the basis (with pre-image e_{13}) gives rise to two new nontrivial syzygies. The syzygy

$$s_{13} = -a_1y^3e_{11} - xe_{13}$$

maps to $-a_2^2x^7 - a_2a_1x^3y^4$, which needs no reduction since its leading monomial is x^7 . The other syzygy, nor those involving x^7 , give new elements. The final standard ideal basis is $\{h_1, h_2, h_3, h_4\} :=$

$$\{xy^2, x^2y + a_1y^4 + a_2x^4, -a_1y^5 + a_2a_1x^2y^4 + a_2^2x^6, -a_2^2x^7 - a_2a_1x^3y^4\}.$$

Using the expressions for the syzygies, and the subsequent reduction, the new basis elements h_3, h_4 can be expressed in the original ones. Algorithm 7.8 collects this information in the map Θ . As a map from M_4 to M_2 , it takes the following form:

$$\Theta = \begin{pmatrix} 1 & 0 & x & -x^2 - a_1y^3 \\ 0 & 1 & -y + a_2x^2 & xy - a_2x^3 \end{pmatrix}.$$

For example, from the fourth column it follows that h_4 is equal to $(-x^2 - a_1y^3)h_1 + (xy - a_2x^3)h_2$. Figure 7.1 depicts the standard basis, and the leading monomial occurring in the ideal.

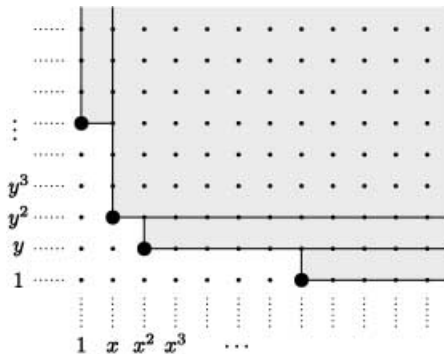


Fig. 7.1 Leading monomials of an ideal, and of its standard basis.

Standard subalgebra basis In the second part of the algorithm, the set $\{g_1, g_2\} = \{a_3(x^2 + y^2), a_4x^4 - a_5y^4\}$ is extended to a standard subalgebra basis. The syzygies involved are called the *syzygies of the first kind*, see Chap. 6. These syzygies may be computed using algorithm 6.24. It begins by computing a Gröbner basis for the ideal

$$\langle z_1 - a_3x^2, z_2 - a_4x^4 \rangle$$

with respect to the elimination term order with $\{z_1, z_2\} < \{x, y\}$. The resulting Gröbner basis is

$$\{a_4z_1^2 - a_3^2z_2, z_1 - a_3x^2\}.$$

Here z_1 and z_2 correspond to the respective generators of the subalgebra. Following algorithm 6.24 we select those basis elements that do not depend on x or y . There is only one such generator, yielding the syzygy $a_4z_1^2 - a_3^2z_2$. Its image under Ψ^g is $2a_3^2a_4x^2y^2 + a_3^2(a_5 + a_4)y^4$, and since its leading monomial is x^2y^2 this element cannot be reduced. A second round gives a larger Gröbner basis, but the only binomial involving only the z_i is the one already considered. The standard subalgebra basis thus becomes $\{g_1, g_2, g_3\} :=$

$$\{a_3(x^2 + y^2), a_4x^4 - a_5y^4, 2a_3^2a_4x^2y^2 + a_3^2(a_5 + a_4)y^4\}.$$

This completes the second stage. The leading monomials of elements of the subalgebra are shown in Fig. 7.2. Those associated with generators are shown as thick bullets.

The data in Fig. 7.2 leaves open the possibility that g_2 , with leading monomial x^4 , is now superfluous, since there is another generator with leading monomial x^2 . Indeed g_2 can be written in terms of the other two, and may be deleted from the generating set, resulting in what could be called a *minimal standard basis*. Such a reduction may shorten subsequent calculations, however we shall not use this.

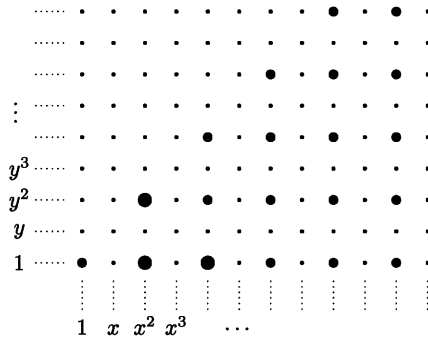


Fig. 7.2 Leading monomials of a subalgebra, and of its standard basis.

Standard left-right tangent space basis Now for the third and final stage. This stage deals with syzygies of the *second* and *third kind*. These involve at least one element of the form $f_i Y(g_1, \dots, g_m)$. If the other element is also of this form, but with a different f_i , we speak of *syzygies of the third kind*. (The same f_i leads to subalgebra-syzygies, or syzygies of the first kind). If the other element is from the ideal $\langle h_i \rangle$, it is a *syzygy of the second kind* – see Chap. 6 for more details.

The syzygies are found by a Gröbner basis calculation. To implement this stage efficiently, all syzygies found are reduced by the normal form algorithm, with nonzero ones being added to the basis, before the syzygies of the extended basis are re-calculated.

The resulting basis may contain superfluous elements. However the reduction of the calculation time is significant, while a larger basis amounts to only a slight overhead for the normal form calculations. The superfluous elements may also be identified and deleted after the standard basis calculation, if desired. The resulting basis may again be called a *minimal standard basis*, in analogy with Gröbner bases (see e.g. [CLO92, p. 90]).

Let's first calculate the syzygies of the third kind, between elements of the ideal and the subalgebra-module. In the notation of lemma 6.29, we need generators $z^\beta e_{2i}$ of the N_{32} -submodule of elements that map into $\text{LM} \langle h_1, h_2, h_3 \rangle$ under $\tilde{\Psi}_2 = \tilde{\Psi}_{N_{32}}$. This is what algorithm 6.34 calculates.

At this point the two generators of the algebra-module are $f_0 = 1$ and $f_1 = x^2 y^2$. The generator $t_2 - f_1$ makes the variable t_2 play the role of f_1 . Of course f_1^2 and higher powers are in general not in T , but we will filter out these higher powers of t_2 later, which is possible by choice of term order. By adding the generators $z_i - g_i$, $i = 1, 2, 3$, we let the variables z_1, z_2, z_3 play the role of subalgebra-generators g_1, g_2, g_3 . Finally, the monomial generators of $\text{LM} \langle h_1, h_2, h_3 \rangle$ are just $\text{LM} h_1, \text{LM} h_2, \text{LM} h_3$ themselves, since they already formed a standard ideal basis. So we have the generators

$$\{t_2 - x^2 y^2, z_1 - a_3 x^2, z_2 - a_4 x^4, z_3 - 2a_3^2 a_4 x^2 y^2, xy^2, x^2 y, -a_1 y^5, -a_2^2 x^7\},$$

a Gröbner basis with respect to the elimination term order with $\{z_i\} \prec \{t_2\} \prec \{x, y\}$ of which is

$$\{z_3, a_4 z_1^2 - a_3^2 z_2, z_2^2, t_2 z_1, t_2 z_2, t_2^2, x z_1 z_2, x t_2, y z_1, y z_2, y t_2 a_3 x^2 - z_1, xy - t_2, y^5\}.$$

We are after the monomials involving only the z_i , and t_2 in degree at most 1. Using algorithm 6.36 we find the following generating set:

$$\{z_3, t_2 z_1, t_2 z_2, z_2^2\}.$$

These monomials correspond to the elements $g_3, f_1 g_1, f_1 g_2$ and g_2^2 with leading monomials $x^2 y^2, x^3 y, x^5 y$ and x^8 respectively, indeed lying in $\text{LM} \langle h_i \rangle$. This gives only one term of the binomial syzygy. The other term is computed by subtracting a multiple of some ideal generator, such that the leading monomial vanishes. The

first syzygy is then e.g. $g_3 - 2a_3^2 a_4 x h_1$, which has leading monomial y^4 and cannot be reduced. This gives a new subalgebra-module generator, $f_2 = a_3^2(a_4 + a_5)y^4$. The second syzygy is $f_1 g_1 - a_3 x h_2$ with leading monomial xy^3 . This one can be reduced, via h_1 . The result has leading monomial x^5 and cannot be reduced further, giving a second new subalgebra-module generator $f_3 = -a_3 x(a_2 x^4 + a_1 y^4)$. The other two elements can be reduced to 0.

Next, we compute the syzygies of the second kind, among elements of the subalgebra-module. The algorithm computing these is described in Proposition 6.26, and leads to two syzygies:

$$a_4 g_1^2 - a_3^2 g_2 \quad \text{and} \quad -a_4 a_3^2(a_4 + a_5)g_3 - 4a_3^4 a_4^2 f_2 g_2.$$

Since after the second stage the algebra generators form a standard basis, the first syzygy, which does not involve the module generators, is guaranteed to reduce to zero. The second must be checked, and turns out also to reduce to zero.

To complete the third stage we have to check all syzygies of the third kind involving the new generators f_2 and f_3 . There are four new syzygies:

$$\begin{aligned} f_2 g_1 - a_3^2 a_4(a_4 + a_5)xy^2 h_1, & \quad f_3 g_1 - \frac{a_3^2 x^7}{a_2} h_4, \\ f_2 g_2 - a_3^2 a_4(a_4 + a_5)x^3 y^2 h_1, & \quad f_3 g_2 - \frac{a_3 a_4}{a_2} x^2 h_4, \end{aligned}$$

which all reduce to 0. The final result is in Fig. 7.3, with the monomials associated to the f_i shown as bullets. Summarizing, we found the following module generators in the final stage:

$$f_1 = xy, \quad f_2 = a_3^2(a_4 + a_5)y^4, \quad f_3 = -a_3(a_2 x^5 + a_1 xy^4).$$

Codimension The monomials *not* in the tangent space T span a complement. In this case a complement is formed by the set

$$\{x, x^3, y, y^2, y^3\},$$

and the codimension of the LR-tangent space is 5.

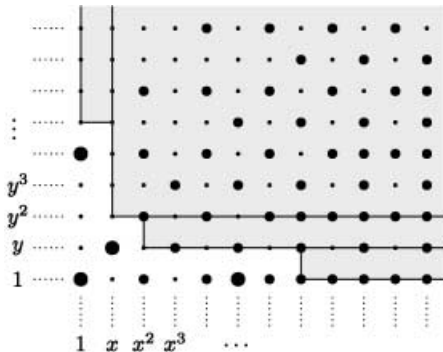


Fig. 7.3 Leading monomials in a LR-tangent space.

Nondegeneracy conditions The set \mathbf{LMT} of leading monomials looks like Fig. 7.3 only if the coefficients associated to those monomials are nonzero. In this case, there are four nondegeneracy-conditions. The following table lists them, together with the related generators, and some monomials that will disappear from \mathbf{LMT} if the condition is not met:

Condition:	Generators:	Monomials:
$a_1 \neq 0$	h_3	y^5
$a_2 \neq 0$	h_4, f_3	x^7, x^5
$a_3 \neq 0$	g_1	x^2
$a_4 + a_5 \neq 0$	f_2	y^4

Note that the leading coefficient $2a_3^2a_4$ of g_3 does not contribute a nondegeneracy-condition, since the associated monomial x^2y^2 (and multiples) is already contained in the ideal-part of \mathbf{LMT} .